Kombinasi Algoritma Vigenere Cipher dan One Time Pad untuk Mengamankan Data Teks

Virushati Hulu[1], Berto Nadeak[2]
[1,2]STMIK Budidarma Medan Jl. Sisingamangaraja No.338 Simpang Limun Medan, Sumatera Utara, Indonesia

ARTICLE INFORMATION

Received: Februari, 20, 2020
Revised: Maret, 6, 2020
Available online: April,9 2020

KEYWORDS

Vigenere cipher, One Time Pad (OTP), kriptografi.

CORRESPONDENCE

Phone: +6285262144030

E-mail: virushati001@gmail.com

A B S T R A K

Perkembangan teknologi turut mempengaruhi tingkat keamanan informasi yang bersifat rahasia. Berbagai pihak yang teidak berkepentingan dapat menggunakan perkembangan teknologi untuk mendapatkan informasi tersebut. Untuk menjaga agar informasi tetap aman, maka digunakan kombinasi algoritma vigenere cipher dan One Time Pad (OTP). Kriptografi ini adalah teknik untuk menyandikan pesan dan menjaga keamanan suatu pesan. Kombinasi ini digunakan sebagai suatu sistem untuk mengamankan data teks, karena data teks mengalami dua proses pengamanan, yaitu proses enkripsi dan proses dekripsi. Enkripsi pada algoritma vigenere cipher dan One Time Pad (OTP) dilakukan sebanyak 2 kali enkripsi dan 2 kali dekripsi, dimana dalam proses ini pertama dilakukan dengan proses enkripsi vigenere cipher hasil cipherteks yang dihasilkan proses vigenere cipher dijadikan sebagai plainteks untuk proses enkripsi One Time Pad (OTP) hasil cipherteks yang dihasilkan dari proses enkripsi (OTP) dan dijadikan sebagai proses dekripsi (OTP) dan hasilnya berupa plainteks yang didapat dari dekripsi (OTP) dan lakukan proses dekripsi dengan vigenere cipher dan gunakan kunci awal dari vigenere cipher untuk mengembalikan cipherteks yang diinput ke plainteks semula sehingga hasilnya berupa plainteks asli dan kembali kebentuk semula..

PENDAHULUAN

Masalah keamanan data ini, menjadi isu yang berkembang pada era teknologi informasi ini. Banyak kejahatan-kejahatan cyber yang pernah kita dengar dari media masa. Pelaku kejahatan memanfaatkan celah keamanan yang ada untuk dimasuki dan melakukan manipulasi. Timbulnya kejahatan-kejahatan tersebut tentu saja memicu para pakar teknologi informasi untuk meningkatkan keamanan dalam pertukaran informasi. Dalam kasus ini Kriptografi memiliki peran dalam membantu meningkatkan keamanan data. Kriptografi adalah ilmu yang mempelajari bagaimana menjaga keamananan suatu pesan (plaintext). Tugas utama Kriptografi adalah untuk menjaga agar pesan dan kunci tetap terjaga kerahasiaannya dari penyadap (attacker). Pada Kritopgrafi dikenal proses enkripsi (penyandian) dan dekripsi (mengembalikan) untuk menghasilkan teks sandi (ciphertext) dan proses dekripsi untuk mengembalikan teks tersandi (plaintext). Penyadap pesan diasumsikan mempunyai akses yang lengkap dalam saluran komunikasi antara pengirim pesan dan penerima pesan. Penyadapan pesan sering terjadi pada komunikasi melalui internet maupun saluran telepon [1].

Algoritma Kriptografi yang dianggap mampu mengamankan data teks seperti Vigenere Cipher dan One Time Pad. Algoritma Vigenere Cipher merupakan salah satu algoritma Kritopgrafi klasik untuk menyadikan suatu plaintext dengan menggunakan tekni subsitusi. Vigenere Cipher menggunakan bujur sangkar untuk melakukan enkripsi dan dekripsi [2]. Kemudian setiap baris di dalam bujusangkar tersebut menyatakan hururf cipherteks yang diperoleh dengan Caesar Cipher, sedangkan algoritma One Time Pad adalah merupakan algoritma yang berisi deretan atau karakter-karakter kunci yang dibangkitkan secara acak. Cipher ini diimplementasikan melalui sebuah kunci yang terdiri dari sekumpulan random karakter-karakter yang tidak berulang. Masing-masing huruf kunci dijumlahkan dengan modulo 255 dengan huruf pada plaintext. Pada One Time Pad, setiap huruf kunci digunakan satu kali untuk satu pesan dan tidak digunakan kembali. Panjang stream karakter kunci sama dengan panjang pesan[3]. Apabila kedua algoritma ini dikombinasikan, keamanan data sangat terjamin dan akurat sehingga orang yang tidak berkepentingan dan tidak memiliki hak akses akan mengalami kesulitan untuk melakukan hal-hal yang tidak dinginkan serta algoritma ini bertahan cukup lama sampai ditemukannya metode untuk memecahkan kedua algoritma tersebut[4].

Berdasarkan latar belakang masalah, maka perumusan masalah dibahas adalah bagaimana prosedur algoritma Vigenere Cipher dan One Time Pad (OTP) dalam mengamankan data teks , bagaimana mengkombinasikan algoritma Vigenere Cipher dan One Time Pad dalam upaya peningkatan keamanan data teks, bagaimana merancang aplikasi pengamanan data teks dengan menerapkan kombinasi algoritma Vigenere Cipher dan One Time Pad dengan batasan masalah adalah prosedur yang akan dibahas adalah prosedur enkripsi dan deksripsi dengan algoritma Vigenere Cipher dan One Time Pad, data teks yang dienkripsi dan didekripsi adalah karakter-karakter teks yang di input secara manual dan termasuk dalam simbol tabel ASCII 255., bahasa pemrograman yang digunakan untuk merancang aplikasi dan untuk mengimplementasikan kombinasi algoritma Vigenere Cipher dan One Time Pad dilakukan dengan bahasa pemrogram Visual Basic 2008.

LANDASAN TEORI

Kombinasi dari sekumpulan objek adalah susunan objek-objek tanpa memperhatikan urutan dari objek dari objek-objek tersebut. Pendekatan dalam penelitian yang mengkombinasikan atau menghubungkan antara metode penelitian kuantitatif dan kualitatif. Hal

ini mencakup landasan filosofis, penggunaan pendekatan kualitatif dan kuantitatif dengan mengkombinasikan kedua pendekatan dalam penelitian

2.1 Kriptografi

Kriptografi *(cryptography)* berasal dari bahasa yunani crypto dan graphia. crypto berarti *secret* (rahasia) dan graphia berarti *writing* (tulisan). Menurut trimologinya, kriptografi adalah ilmu seni untuk menjaga keamanan pesan ketika pesan dikirim dari suatu tempat ketempat lain [5]. Encryption adalah transformasi data kedalam bentuk yang tidak dapat terbaca tanpa sebuah kunci tertentu [1]. Tujuannya adalah untuk meyakinkan privasi dengan menyembunyikan informasi dan orang-orang yang tidak ditujukan, bahkan mereka yang memiliki akses ke data terenkripsi. Dekripsi merupakan kebalikan dan enkripsi, yaitu transformasi data terenkripsi kembali ke bentuknya semula.

Enkripsi dan dekripsi pada umumnya membutuhkan penggunaan sejumlah informasi rahasia, disebut sebagai kunci. Beberapa mekanisme enkripsi, kunci yang sama digunakan baik untuk enkripsi dan dekripsi untuk mekanisme yang lain, kunci yang digunakan untuk enkripsi dan dekripsi berbeda. Dua tipe dasar dan teknologi kriptografi adalah *symmetric key (secret/private key) cryptography* dan *asymmetric (public key) cryptography. Symmetric keycryptography*, baik pengirim maupun penerima memiliki kunci rahasia yang umum. *Asymmetric key cryptography*, pengirim dan penerima masing-masing berbagi kunci publik dan privat[6].

2.2 Algoritma Kriptografi

Adapun yang menjadi algoritma kriptografi berdasarkan perkembangannya adalah sebagai berikut [1]:

1. Kriptografi klasik
   Kriptografi klasik merupakan suatu algoritma yang menggunakan satu kunci untuk mengamankan data. Teknik ini sudah digunakan beberapa abad yang lalu.
   Dua teknik dasar yang biasa digunakan pada algoritma jenis ini adalah sebagai berikut:
   a. Teknik substitusi penggantian setiap karakter teks asli dengan karakter lain.
   b. Teknik transposisi dilakukan dengan menggunakan permutasi karakter.
2. Kriptografi modern
   Kriptografi modern mempunyai kerumitan yang sangat kompleks karena dioperasikan menggunakan komputer. Hal ini akan dibahas lebih detail pada bagian lain.

2.3 Vigenere Cipher

Vigenere chiper merupakan salah satu contoh chiper abjad-majemuk *(polyalphabetic substitution chiper*). Chiper abjad-majemuk akan mengganti setiap karakter pada plainteks dengan karakter lain yang mungkin berbeda-beda pada chiperteksnya. Vigenere chiper menggunakan bujur sangkar vigenere untuk melakukan enkripsi [7]. Setiap baris di dalam bujur sangkar menyatakan huruf-huruf chiperteks yang diperoleh dengan caesar chiper, di mana jauh pergeseran huruf plainteks ditentukan oleh nilai desimal dari huruf kunci tersebut (A = 0, B = 1, C = 3, Z = 25).

Melakukan enkripsi dengan vigenere chiper, lakukan pada bujur sangkar vigenere sebagai berikut tarik garis vertikal dari huruf plainteks ke bawah, lalu tarik garis mendatar dari huruf kunci ke kanan. Perpotongan kedua garis tersebut menyatakan huruf chiperteksnya. Pada vigenere chiper, jika panjang kunci lebih pendek daripada panjang plainteks, maka kunci tersebut akan diulang penggunaannya [8]

Contoh penggunaan Vigenere chiper

P : SAYASUKAKRIPTOGRAFI

K : MUSIKMUSIKMUSIKMUSI

C : EUQICGESSBUJLWQDUXQ

Contoh di atas, plainteks “SAYA SUKA KRIPTOGRAFI” dienkripsi dengan kunci “MUSIK” menghasilkan chiperteks “E U Q I C G E S S B U J L W Q D U X Q”.

Perhatikan bahwa huruf A pada plainteks disubstitusi dengan huruf yang berbeda-beda pada chiperteks, yakni U, I, S. Hal inilah yang menyebabkan vigenere chiper termasuk chiper abjad-majemuk. Aturan enkripsi pada vigenere chiper bisa dinyatakan juga sebagai penjumlahan modulo 26 dari satu karakter plainteks dengan satu karakter kunci[1].

Ci = (Pi + Ki) mod 26

dimana

Pi : karakter plainteks

Ki : karakter kunci

Ci : karakter chiperteks

Dekripsi pada vigenere chiper dilakukan dengan cara yang berkebalikan, yaitu dengan cara menarik garis horizontal dari huruf kunci sampai ke huruf chiperteks yang dituju, lalu dari huruf cipherteks tarik garis vertikal ke atas sampai ke huruf plainteks atau bisa juga dinyatakan dalam persamaan Pi = (Ci- Ki) mod 26 [1].

Kekuatan algoritma vigenere chiper ini adalah dapat mencegah frekuensi huruf-huruf di dalam chiperteks yang memiliki pola tertentu yang sama, seperti yang terjadi pada chiper abjad tunggal. Chiper abjad tunggal, huruf yang paling sering muncul di chiperteks merupakan substitusi dari huruf yang paling sering muncul di plainteks. Karena itu, dengan teknik analisis frekuensi, kriptanalis bisa dengan mudah menebak huruf tersebut. Namun, pada vigenere chiper hal tersebut tidak bisa dilakukan karena satu macam huruf pada plainteks mungkin dienkripsi menjadi beberapa macam huruf pada chiperteks, seperti pada contoh sebelumnya.

Vigenere chiper memungkinkan perulangan huruf atau pasangan huruf pada plainteks terjadi juga pada chiperteksnya. Hal ini dikarenakan kunci yang digunakan untuk melakukan enkripsi juga diulang. Akibatnya, bagian plainteks dan bagian kunci tertentu bisa "berpasangan" lebih dari satu kali [9].
Contoh :
P : SAYACINTAPACARSAYA
K : KASIHKUKASIHKUKASI
C : CAQIJSHDAHIJKLCAQI

Terlihat pada contoh di atas, SAYA dienkripsi menjadi kriptografi yang sama, yaitu CAQI. Namun, perlu diperhatikan bahwa kasus seperti ini tidak selalu demikian, misalnya pada contoh berikut ini :
P : SAYACINTAPACARSAYA
K : SAYANGKUSAYANGKUSA
C : KAWAPOXNSPYCNXCUQA
Contoh di atas, SAYA tidak dienkripsi menjadi kriptografi yang sama. Sifatnya yang mungkin untuk menghasilkan kriptografi yang sama terhadap bagian plainteks yang sama ini menjadi kelemahan vigenere chiper.

2.4 One Time Pad
Sandi one time pad merupakan sandi yang mencapai kerahasiaan sempurna *(perfect secrecy)* yaitu menghasilkan teks sandi yang tidak memiliki hubungan statistik terhadap teks asli sehingga analisis statistik tidak dapat dilakukan. Sandi one time pad bekerja dengan menghasilkan kunci yang berbeda setiap karakternya untuk semua teks asli, kunci dibangkitkan secara acak dan deretan kunci digunakan hanya sekali saja[5]

Shannon membuktikan apabila sandi one time pad diterapkan secara benar maka sandi one time pad mencapai kerahasiaan sempurna *(perfect secrecy),* (Shannon, 1949). Shannon mendefinisikan sebuah sistem sandi mencapai perfect secrecy bila pasangan teks asli dan teks sandi tidak memiliki hubungan statistik sehingga sulit bagi penyerang untuk melakukan analisis sandi atau analisis statistik. Kelemahan utama sandi one time pad adalah ketidakpraktisan. Kunci pada sandi one time pad memiliki panjang sama dengan panjang teks asli [6].

Aturan enkripsi yang digunakan persis sama seperti pada kode vigenere [10]
1. Enkripsi j = (Pi + kJ) mod 26
2. Dekripsi ci = (Pi-fa) mod 26 Bila diketahui teks asli "ONETIMEPAD" dengan kunci "TBFRGFARFM" diasumsikan A=0, B=1, Z 25, maka akan didapat teks kode "IPKLPSFHGQ" yang mana diperoleh sebagai berikut :
   (0 + T) mod 26 = I
   (N + B) mod 26 = P
   (E + F) mod 26 = K
   (T + R) mod 26 = L
   (1 + G) mod 26 = P
   (M + F) mod 26 = S
   (E + A) mod 26 = F
   (P + R) mod 26 = H
   (A + F) mod 26 = G
   (D +M) mod 26 = Q

Sistem OTP tidak dapat dipecahkan karena beberapa alasan
1. Barisan kunci acak + teks asli yang tidak acak teks kode yang seluruhnya acak.
2. Mendekripsi teks kode dengan beberapa kunci berbeda dapat menghasilkan teks asli yang bermakna sehingga kriptanalis tidak punya cara untuk menentukan teks asli mana yang benar.

Contoh :

Kriptanalis mencoba rnendekripsi teks kode IPKLPSFHGQ
Kriptanalis mencoba kunci : POYY AEAAZX
Teks asli yang dihasilkan : SALMONEGGS
Bila ia mencoba kunci : BXFGBMTMXM
Teks asli yang dihasilkan : GREENFLUID

Berdasarkan contoh tersebut sudah jelas bahwa kriptanalis akan bingung atau rnendapatkan teks asli yang salah, bukan "ONETIMEPAD". Meskipun OTP merupakan suatu algoritma yang sempurna dan aman, tetapi dalam praktik OTP jarang digunakan karena sedikit rumit yang disebabkan oleh panjang kunci = panjang pesan, sehingga timbul masalah penyimpanan kunci, pendistribusian kunci dan masalah pengiriman kunci karena kunci dibangkitkan secara acak, maka tidak mungkin pengirim dan penerima membangkitkan kunci yang sama secara simultan. OTP hanya dapat digunakan jika tersedia saluran komunikasi alternatif yang cukup aman untuk mengirim kunci. Saluran ini pada umumnya cukup aman dan lambat.

HASIL DAN PEMBAHASAN

Keamanan merupakan aspek yang paling penting dalam informasi. Sebagian orang tidak ingin data ataupun informasi yang dikirimkan atau ditujukan kepada orang lain diketahui oleh pihak yang tidak berhak untuk menerima data tersebut. Beberapa masalah yang sering ditemui dalam hal keamanan data teks, dimana masih banyak terjadinya kegagalan pada keamanan data teks seperti penyadapan dan perubahan terhadap isi data teks asli. Masalah tersebut dapat terjadi dikarenakan kurang rumitnya penerapan algoritma pada sistem keamanan data teks.

Sistem keamanan pada data teks jika diterapkan kombinasi dari beberapa algoritma kriptografi, maka akan meminimalisir dan mencegah terjadinya penyadapan atau pembobolan data teks. Salah satu algoritma kriptografi yang merupakan kombinasi dari dua algoritma dalam proses enkripsinya adalah *vigenere cipher* dan *one time pad*. Algoritma *vigenere cipher* dan *one time pad* termasuk dalam kategori super enkripsi, karena enkripsinya menggunakan kombinasi. Tingkat ketergantungan cipherteks terhadap kunci pada algoritma ini juga sangat tinggi. Salah satu huruf saja, maka akan berakibat kesalahan pada cipherteks dan untuk menambah tingkat kerumitan dalam pemecahan cipherteks disarankan menggunakan kunci yang berbeda antara satu dengan yang lainnya.

3.1. Analisa dan Logika Metode

Enkripsi pada algoritma *vigenere cipher* dan *One Time Pad* (OTP) dilakukan sebanyak 2 kali enkripsi dan 2 kali dekripsi, kombinasi dalam algoritma ini menggunakan algoritma *vigenere cipher* dan *One Time Pad* (OTP). Adapun langkah-langkah proses enkripsi dari algoritma *vigenere cipher* dan *One Time Pad* (OTP) adalah sebagai berikut:

1. Lakukan proses enkripsi dengan *vigenere cipher* dengan cara :
   a. Tentukan plainteks (pesan teks asli) yang akan dienkripsi.
   b. Buat kunci untuk proses enkripsi.
   c. Buat tabel enkripsi, dimana jumlah kolom yang dibentuk sama dengan jumlah plainteks yang digunakan dan tulis plainteks dengan orientasi baris.
   d. Cari cipherteks dengan rumus $C_i = (P_i + K_i)$ Mod 255, lakukan sampai pada plainteks terakhir. Setelah itu untuk mendapatkan karakter cipherteks ubah bilangan desimal ke karakter sesuai dengan kode ASCII 8 *bit* atau 256 karakter.
   e. Hasilnya berupa cipherteks yang didapatkan dari hasil enkripsi *vegenere cipher*.
   f. Cipherteks yang dihasilkan dari proses enkripsi *vigenere cipher* dijadikan sebagai plainteks pada proses enkripsi *One Time Pad* (OTP).
2. Lakukan proses enkripsi dengan algoritma *one time pad* dengan cara :
   a. Menentukan kunci, dimana hasil dari cipherteks *vegenere cipher* dijadikan plainteks pada proses enkripsi dengan algoritma *one time pad* panjang kunci *one time pad* sama dengan panjang karakter cipherteks *vigenere cipher*.
   b. Buat tabel untuk merubah karakter plainteks dan karakter kunci ke dalam bilangan desimal sesuai dengan kode ASCII 8 *bit* atau 255 karakter dan tempatkan plainteks sesuai dengan kuncinya masing-masing.
   c. Cari cipherteks dengan rumus $C_i = (P_i + K_i)$ Mod 255, lakukan sampai pada plainteks terakhir. Setelah itu untuk mendapatkan karakter cipherteks ubah bilangan desimal ke karakter sesuai dengan kode ASCII 8 *bit* atau 255 karakter.
   d. Cipherteks yang dihasilkan pada proses enkripsi *One Time Pad* (OTP) inilah yang menjadi cipherteks yang digunakan.

Proses dekripsi pada algoritma ini dilakukan sebanyak 2 kali dekripsi. Kunci yang digunakan untuk melakukan proses dekripsi sama dengan kunci yang digunakan pada saat proses enkripsi, karena jenis kunci yang digunakan pada algoritma *vigenere cipher* dan *One Time Pad* (OTP) ini adalah jenis kunci simetrik.
Langkah-langkah untuk proses dekripsi dari algoritma *vigenere cipher* dan *One Time Pad* (OTP) adalah sebagai berikut:

1. Lakukan proses dekripsi dengan algoritma OTP dengan cara :
   a. Hasil cipherteks yang didapat dari hasil enkripsi OTP dijadikan sebagai plainteks untuk proses dekripsi dengan algoritma OTP dan input kunci OTP seperti yang digunakan pada proses enkripsi OTP.
   b. Buat tabel enkripsi, dimana jumlah kolom yang dibentuk sama dengan jumlah plainteks yang digunakan dan tulis plainteks dengan orientasi baris.
   c. Buat tabel untuk merubah karakter plainteks dan karakter kunci ke dalam bilangan desimal sesuai dengan kode ASCII 8 *bit* atau 255 karakter dan tempatkan plainteks sesuai dengan kuncinya masing-masing.
   d. Cari cipherteks dengan rumus $C_i = (P_i - K_i)$ Mod 255, lakukan sampai pada plainteks terakhir. Setelah itu untuk mendapatkan karakter cipherteks ubah bilangan desimal ke karakter sesuai dengan kode ASCII 8 *bit* atau 255 karakter.
   e. Hasilnya berupa plainteks yang didapatkan dari dekripsi *one time pad*.

2. Lakukan proses dekripsi dengan algoritma *vigenere cipher*
   a. Gunakan hasil proses dekripsi *one time pad* untuk dijadikan sebagai cipherteks pada proses dekripsi *vigenere cipher*.
   b. Gunakan kunci awal dari *vigenere cipher* untuk mengembalikan cipherteks yang dinput ke plainteks semula.
   c. Buat tabel untuk merubah karakter cipherteks dan karakter kunci ke dalam bilangan desimal sesuai dengan kode ASCII 8 *bit* atau 255 karakter dan tempatkan cipherteks sesuai dengan kuncinya masing-masing.
   d. Cari plainteks dengan rumus $P_i = (C_i - K_i)$ Mod 255, lakukan sampai pada cipherteks terakhir. Setelah itu untuk mendapatkan karakter plainteks ubah bilangan desimal ke karakter sesuai dengan kode ASCII 8 *bit* atau 256 karakter.
   e. Hasilnya berupa plainteks asli dengan kembali kebentuk semula.

3.2. Penerapan Algoritma *Vigenere Cipher* dan *One Time Pad*

Penerapan algoritma *vigenere cipher* dan *One Time Pad* (OTP) dalam keamanan data teks dilakukan dengan cara mengenkripsi data teks yang akan diamankan agar tidak dapat dibaca informasinya oleh orang yang tidak berhak dengan menggunakan kunci yang berbeda antara algoritma *vigenere cipher* dan algoritma *One Time Pad* (OTP), kemudian agar penerima data dapat mengerti informasi yang telah dikirimkan, maka dilakukan proses dekripsi dengan menggunakan kunci yang digunakan pada saat enkripsi atau kunci yang telah disepakati oleh kedua belah pihak (pengirim dan penerima). Contoh penerapan algoritma *vigenere cipher* dan OTP sebagai berikut:

Sebagai contoh penerapan algoritma *vigenere cipher* dan *one time pad*, jika plainteks adalah V I R U S H A T I dan kunci adalah A B C, maka proses enkripsi yang terjadi adalah sebagai berikut:
Plainteks : V I R U S H A T I
Key : A B C

Pembentukan tabel untuk merubah karakter plainteks dan karakter kunci dalam bilangan desimal sesuai dengan kode ASCII 8 bit atau 256 karakter dan tempatkan plainteks sesuai dengan kuncinya masing-masing

P. (V.C)	V	I	R	U	S	H	A	T	I
Dec.	86	73	82	85	83	72	65	84	73
Key. (V.C)	A	B	C	A	B	C	A	B	C
Dec	65	66	67	65	66	67	65	66	67

Mencari cipherteks dengan menggunakan rumus dan ubah hasil dari perhitungan yang didapat (bilangan desimal) ke bentuk karakter sesuai dengan kode ASCII 8 bit atau 256 karakter

1. Proses enkripsi dengan *vigenere cipher*

C1 = (P1 + Ki) Mod 255
= (V + K) Mod 255
= (86 + 65) Mod 255
= 151 Mod 255
= 151 (—)

C2 = (P2 + Ki) Mod 255
= (I + K) Mod 255
= (73 + 66) Mod 255
= 139 Mod 255
= 139 (‹)

C3 = (P3 + Ki) Mod 255
= (R + K) Mod 255
= (82 + 67) Mod 255
= 149 Mod 255
= 149 (•)

C4 = (P4 + Ki) Mod 255
= (U + K) Mod 255
= (85 + 65) Mod 255
= 150 Mod 255
= 150 (_)

C5 = (P5 + Ki) Mod 255
= (S + K) Mod 255
= (83 + 66) Mod 255
= 149 Mod 255
= 149 (•)

C6 = (P6 + Ki) Mod 255
= (H + K) Mod 255
= (72 + 67) Mod 255
= 139 Mod 255
= 139 (‹)

C7 = (P7 + Ki) Mod 255
= (A + K) Mod 255
= (65 + 65) Mod 255

= 130 Mod 255
= 130 (,)
C8 = (P8 + Ki) Mod 255
= (T + K) Mod 255
= (84 + 66) Mod 255
= 150 Mod 255
= 150 (_)
C9 = (P6 + Ki) Mod 255
= (I + K) Mod 255
= (73 + 67) Mod 255
= 140 Mod 255
= 140 (Œ)

Ciperteks yang dihasilkan dari *vigenere cipher* adalah :

—	‹	•	_	•	‹	,	_	Œ

2. Proses enkripsi dengan OTP

Proses algoritma enkripsi algoritma *one time pad* ini diawali dengan menentukan plainteks dan kunci, dimana plainteksnya diambil dari hasil enkripsi dari *vegenere cipher* dan kuncinya ditentukan sendiri dan disesuaikan dengan pajang plainteksnya.

Plainteks : — ‹ • _ • ‹ , _ Œ
Key : E R V I L Q G U T

Pembentukan tabel untuk merubah karakter plainteks dan karakter kunci dalam bilangan desimal sesuai dengan kode ASCII 8 bit atau 256 karakter dan tempatkan plainteks sesuai dengan kuncinya masing-masing

Tabel 1. Pembentukan Kode ASCII

P. (V.C)	—	‹	•	_	•	‹	,	_	Œ
Dec.	151	139	149	150	149	139	130	150	140
Key. (OTP)	E	R	V	S	L	Q	G	U	T
Dec.	69	82	86	83	76	88	71	85	84

Mencari cipherteks dengan menggunakan rumus (Pi+Ki) Mod 255 dan ubah hasil dari perhitungan yang didapat (bilangan desimal) ke bentuk karakter sesuai dengan kode ASCII 8 *bit* atau 256 karakter

(151 + 69) mod 255 = 220 = Ü
(139 + 82) mod 255 = 221 = Ý
(149 + 86) mod 255 = 235 = ë
(150 + 83) mod 255 = 233 = é
(149 + 76) mod 255 = 225 = á
(139 + 88) mod 255 = 227 = ã
(130 + 71) mod 255 = 201 = É
(150 + 85) mod 255 = 235 = ë
(140 + 84) mod 255 = 224 = à

Ciperteks yang dihasilkan dari *One Time Pad* (OTP) adalah :

Ü	Ý	ë	é	á	ã	É	ë	à

Cipherteks yang dihasilkan inilah yang menjadi sebagai cipherteks akhir dengan dua algoritma yaitu dekripsi dengan OTP dan dekripsi *vigenere cipher* dan kunci yang digunakan pada proses dekripsi adalah sama seperti yang digunakan pada proses enkripsi.

1. Proses dekripsi dengan OTP

Membuat tabel untuk merubah karakter plainteks dan karakter kunci dalam bilangan desimal sesuai dengan kode ASCII 8 bit atau 256 karakter dan tempatkan plainteks sesuai dengan kuncinya masing-masing

Tabel 2. Pembentukan Kode ASCII

P. (OTP)	Ü	Ý	ë	é	á	ã	É	ë	à
Dec.	220	221	235	233	225	227	201	235	224
Key (OTP)	E	R	V	S	L	X	G	U	T
Dec.	69	82	86	83	76	88	71	85	84

Mencari plainteks dengan menggunakan algoritma dekripsi OTP dengan rumus (Pi-Ki) Mod 255 kemudian ubah hasil dari perhitungan yang didapat (bilangan desimal) ke bentuk karakter sesuai dengan kode ASCII 8 *bit* atau 256 karakter.

(220 - 69) mod 255 = 151 = —
(221 - 82) mod 255 = 139 = ‹
(235 - 86) mod 255 = 149 = •
(233 - 83) mod 255 = 150 = _
(225 - 76) mod 255 = 149 = •
(227 - 88) mod 255 = 139 = ‹
(201 - 71) mod 255 = 130 = ,
(235 - 85) mod 255 = 150 = _
(224 - 84) mod 255 = 140 = Œ

Plainteks yang dihasilkan dari proses dekripsi OTP

—	‹	•	_	•	‹	,	_	Œ

Plainteks hasil dekripsi OTP ini akan dijadikan sebagai plainteks pada proses dekripsi *vigenere cipher*

2. Proses dekripsi dengan *vigenere cipher*

Membuat tabel untuk melakukan konversi karakter plainteks dan karakter ke dalam bilangan desimal sesuai dengan kode ASCII 8 *bit* atau 256 karakter dan tempatkan plainteks sesuai dengan kuncinya masing-masing.

Tabel 3. Pembentukan Kode ASCII

P. (OTP)	—	‹	•	_	•	‹	,	_	Œ
Dec.	151	139	149	150	149	139	130	150	140
Key.(V.C)	A	B	C	A	B	C	A	B	C
Dec	65	66	67	65	66	67	65	66	67

Mencari plainteks dengan menggunakan rumus dekripsi *vigenere cipher* kemudian lakukan sampai pada cipherteks terakhir. Setelah itu untuk mendapatkan karakter plainteks ubah bilangan desimal ke karakter sesuai dengan kode ASCII 8 *bit* atau 256 karakter

P1 = (C1 - Ki) Mod 255
= (— - K) Mod 255
= (151 - 65) Mod 255
= 86 Mod 255
= 86 (V)

P2 = (C2 - Ki) Mod 255
= (‹ - K) Mod 255
= (139 - 66) Mod 255
= 73 Mod 255
= 73 (I)

P3 = (C3 - Ki) Mod 255
= (• - K) Mod 255
= (149 - 67) Mod 255
= 82 Mod 255
= 82 (R)

P4 = (C4 - Ki) Mod 255
= (_ - K) Mod 255
= (150 - 65) Mod 255
= 85 Mod 255
= 85 (U)

P5 = (C5 - Ki) Mod 255
= (• - K) Mod 255
= (149 - 66) Mod 255
= 83 Mod 255
= 83 (S)

P6 = (C6 - Ki) Mod 255
= (‹ - K) Mod 255
= (139 - 67) Mod 255
= 72 Mod 255
= 72 (H)

P7 = (C7 - Ki) Mod 255
= (, - K) Mod 255
= (130 - 65) Mod 255
= 65 Mod 255

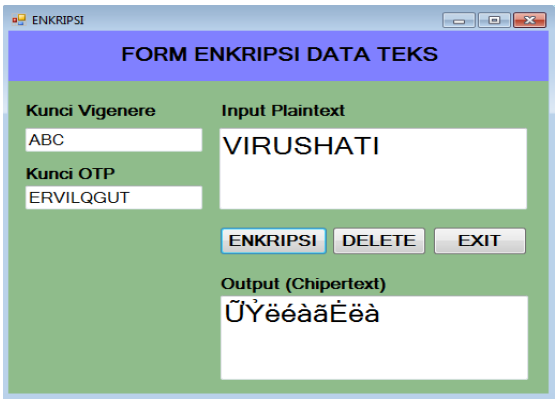
= 65 (A)
P8 = (C8 - Ki) Mod 255
= (_ - K) Mod 255
= (150 - 66) Mod 255
= 84 Mod 255
= 84 (T)
P9 = (C9 - Ki) Mod 255
= (Œ - K) Mod 255
= (140 - 67) Mod 255
= 73 Mod 255
= 73 (I)

Hasil berupa plainteks dari penggabungan algoritma *vigenere cipher* dan *one time pad* kembali ke plainteks semula.

V	I	R	U	S	H	A	T	I

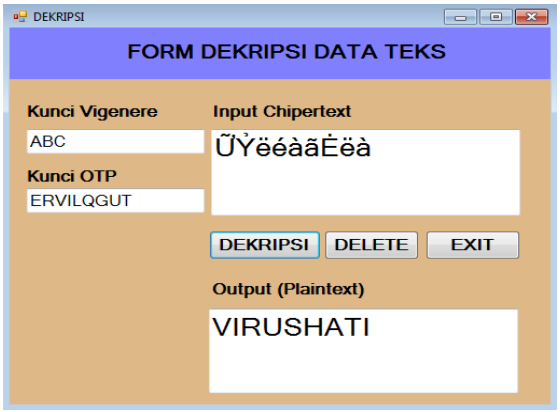
Berdasarkan proses enkripsi dan dekripsi, maka dapat disimpulkan bahwa: Plainteks = VIRUSHATI
Cipher *vigenere cipher* dengan kunci ABC = — ‹ • _ • ‹ , _ Œ
Cipher OTP dengan kunci ERVSLXGUT = Ü Ý ë é á ã É ë à

Sistem dibangun menggunakan Microsoft Visual Studio 2008, berikut merupakan implementasi sistem dari langkah pertama hingga penentuan solusi. Berikut merupakan tampilan *menu* utama yang berfungsi untuk memanggil *form-form* yang ada dalam program. *Form* ini berfungsi untuk melakukan proses perubahan data teks asli (plainteks) menjadi teks sandi (cipherteks). Berikut merupakan tampilan *form* enkripsi data teks.



Gambar 1. Tampilan Form Enkripsi

Form ini berfungsi untuk melakukan proses dekripsi, dimana *user* harus menginputkan cipherteks dan kunci yang didekripsi. Berikut ini tampilan *form* dekripsi.



Gambar 2. Tampilan *Form* Dekripsi

Berikut ini merupakan tampilan dari *form about* yang berfungsi untuk menampilkan informasi penulis.

KESIMPULAN

Berdasarkan penjelasan sebelumnya maka dapat disimpulkan sebagai berikut:

1. Proses penyandian data dengan algoritma vigenere cipher dan One Time Pad (OTP) berhasil digunakan untuk menyembunyikan data rahasia kedalam bentuk simbol-simbol yang tidak bisa dibaca.

2. Teknik enkripsi dan dekripsi data teks ini telah berhasil meningkatkan keamanan data teks dengan menggabungkan kedua metode dengan data teks yang terenkripsi ini tidak akan dapat dibaca, jika tidak didekripsikan dengan kunci yang benar.
3. Program aplikasi ini dapat mengubah data teks menjadi data teks sandi, sehingga mempersulit pihak-pihak yang tidak berkepentingan untuk mengetahui data teks asli.

DAFTAR PUSTAKA

[1] R. Munir, "Kriptografi," *Inform. Bandung*, 2006.
[2] F. Anita, "Implementasi Algoritma Modular Multiplication Based Block Cipher Dalam Mengamankan Data Teks," *MEANS (Media Inf. Anal. dan Sist.*, vol. 3, no. 2, pp. 121–125, 2018.
[3] S. Sari, "Perancangan Aplikasi Pengamanan Pesan Teks Menggunakan Algoritma One Time Pad Berbasis Android," *KAKIFIKOM (Kumpulan Artik. Karya Ilm. Fak. Ilmu Komputer)*, vol. 01, no. 1, pp. 23–26, 2019.
[4] B. Silaban and T. Limbong, "Aplikasi Pembelajaran Pengenalan Kriptografi Algoritma Affine Cipher Dan Vigenere Cipher Menggunakan Metode Computer Assisted Instruction," *Media Inf. Anal. dan Sist.*, vol. 2, no. 2, pp. 14–20, 2017.
[5] D. Ariyus, "Kriptografi keamanan data dan komunikasi," *Yogyakarta Graha Ilmu*, 2006.
[6] N. E. Saragih, "IMPLEMENTASI ALGORITMA ONE TIME PAD PADA PESAN," *J. Ilm. Matrik*, vol. 20, no. 1, pp. 31–40, 2018.
[7] L. Endah Pratiwi, R. Marwati, and I. Yusnitha, "PROGRAM APLIKASI KRIPTOGRAFI PENYANDIAN ONE TIME PAD MENGGUNAKAN SANDI VIGENERE."
[8] H. Sahara, "Implementasi Pengamanan Pesan Chatting menggunakan Metode Vigenere Cipher dan Cipher Block Chainning," *MEANS (Media Inf. Anal. dan Sist.*, vol. 3, no. 2, pp. 173–178, 2018.
[9] R. Munir, "Algoritma & Pemrograman dalam Bahasa Pascal dan C Edisi Revisi," *Andi Yogyakarta*, 2011. [Online]. Available: https://openlibrary.telkomuniversity.ac.id/pustaka/21198/algoritma-pemrograman-dalam-bahasa-pascal-dan-c-edisi-revisi.html. [Accessed: 19-Feb-2020].
[10] F. Diani and Y. Widhiyasana, "Enkripsi SMS dengan Menggunakan One Time Pad (OTP) dan Kompresi Lempel-Ziv-Welch (LZW)," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 7, no. 3, pp. 3–8, 2018.